Distributional Off-Policy Evaluation in Reinforcement Learning

Zhengling Qi[*], Chenjia Bai[†], Zhaoran Wang[‡], Lan Wang, [§]

Abstract

In the existing literature of reinforcement learning (RL), off-policy evaluation is mainly focused on estimating a value (e.g., an expected discounted cumulative reward) of a target policy given the pre-collected data generated by some behavior policy. Motivated by the recent success of distributional RL in many practical applications, we study the distributional off-policy evaluation problem in the batch setting when the reward is multi-variate. We propose an offline Wasserstein-based approach to simultaneously estimate the joint distribution of a multivariate discounted cumulative reward given any initial state-action pair in the setting of an infinite-horizon Markov decision process. Finite sample error bound for the proposed estimator with respect to a modified Wasserstein metric is established in terms of both the number of trajectories and the number of decision points on each trajectory in the batch data. Extensive numerical studies are conducted to demonstrate the superior performance of our proposed method.

Keywords: Distributional reinforcement learning; Off-policy Evaluation; Multivariate reward; Wasserstein metric; Infinite-horizon Markov decision processes

1 Introduction

The primary goal of batch reinforcement learning (RL) is to leverage pre-collected data to learn optimal sequential decision rules for maximizing the cumulative rewards one can receive. In recent years, we have seen many successful applications of deploying batch RL techniques in various scientific fields. For example, in mobile health study, batch RL is used to construct individualized sequences of treatments for promoting healthy behavior (Liao et al., 2020), as an example of precision medicine (Kosorok and Laber, 2019). In

[*]Department of Decision Sciences, The George Washington University
[†]Department of Computer Science, The Harbin Institute of Technology
[‡]Departments of Industrial Engineering, The Northwestern University
[§]Department of Management Science, The University of Miami

industrial organization, econometricians study dynamic discrete choice models (Rust, 1987) for understanding the rationality behind decision makers, which is often called inverse RL in the machine learning community. Batch RL has also been widely used in improving business operations such as dynamic pricing (Liu et al., 2019), digital marketing (Thomas et al., 2017), finance and logistics (Hubbs et al., 2020), etc. For an overview of batch RL, we refer to Levine et al. (2020) and the references therein. While batch RL has demonstrated its great potentials in practice without interacting with task environment, which is very suitable for high-stake domains, one essential challenge is how to make use of the historical data to evaluate a target policy when there is a distributional mismatch with the data generating process.

Motivated by this, we study the off-policy evaluation problem, which is considered as a building block for many (batch) RL algorithms. Most existing literature are focused on estimating the value (i.e., a single-dimensional expected cumulative discounted reward) of a target policy using the batch data. However, in a variety of applications, criteria other than the value may be more sensible and multiple rewards or risk factors are often observed in the decision process. For example, when implementing a policy (e.g., a trading strategy) in the financial market, the corresponding volatility is of importance for the risk control, in addition to the expected return (Wang and Zhou, 2020). In medical studies, doctors/clinicians often investigate on how a new treatment strategy can improve patients on the lower tails (e.g., risk of adverse events), besides maximizing the clinical benefits (Wang et al., 2018). The quantiles or conditional value at risk of multiple outcomes under a certain treatment rule is then of the great interest. Therefore, it is essential to estimate the distributional behavior of any target policy using the pre-collected data to fully characterize its performance from multiple lens (i.e., multiple rewards/risk measures).

Motivated by these, in this paper, we study the general estimation problem on the entire distribution of a multi-variate cumulative discounted reward of any target policy given any

initial state-action pair under the framework of infinite-horizon Markov decision processes (MDPs). Different from the conventional distribution estimation problem, realizations from the target random vector is not directly observed since the cumulative discounted reward is defined on the infinite-horizon. Thanks to the distributional Bellman equation (e.g., Chung and Sobel (1987)), which is a special form of integral equation (Kress et al., 1989) and closely related to the conditional moment model (Ai and Chen, 2003; Newey and Powell, 2003), we formulate our estimating problem as a minimax two-player game. Our estimator is also inspired by the spirit of recently proposed generative adversarial neural networks (Goodfellow et al., 2014, GANs) and later Wasserstein-GANs (Arjovsky et al., 2017). Specifically, we propose a Wasserstein GAN-based approach to estimate the distribution of multi-variate discounted cumulative rewards given any target policy in the batch setting. Our approach is mainly driven by the contraction property of the distributional Bellman operator with respect to the (modified) Wasserstein metric, i.e., supreme Wasserstein metric. We then provide a thorough statistical analysis on the proposed estimating procedure. In particular, under some technical conditions, we establish the finite-sample error bound for our estimated distribution and show it converges to the true one in the sense of (supreme) 1-Wasserstein metric as long as N or T goes to infinity. Here N refers to the number of trajectories and T as the number of decision points on each trajectory in our batch data. To the best of our knowledge, this is the first established non-asymptotic bound for estimating the distribution of cumulative discounted reward under the batch setting in the infinite-horizon MDP. Our theoretical result does not require the data come from the stationary distribution and thus can be broadly applied in practice. This relies on the new concentration inequality shown in Lemma 3.1 in the appendix, which may be of independent interest. Lastly, we conduct an extensive numerical study to demonstrate the superior performance of our estimation method in capturing the joint distributional behavior of a multi-variate cumulative discounted reward, which we believe will serve as

a foundation for many existing distributional policy optimization methods and moreover designing new algorithms for optimizing general objectives in the sequential decision making process. See several motivating examples in Section 2.2.

1.1 Related Work

Recently there is a surging interest in studying the distributional RL, due to its empirical success in Atari benchmark (Bellemare et al., 2017; Dabney, Rowland, Bellemare and Munos, 2018; Dabney, Ostrovski, Silver and Munos, 2018; Yang et al., 2019). Instead of estimating the value of a policy, distributional RL targets on the whole distribution when searching for the optimal policy that maximizes the value. While it has demonstrated its great potentials (See numerical comparisons in Fu et al. (2021); Agarwal et al. (2021)), theoretical investigation is rather limited. To the best of our knowledge, Jaquette (1973) and Chung and Sobel (1987) are among the first few papers studying the theoretical property of the distributional Bellman operator. Recently, Bellemare et al. (2017) studied the property of distributional RL when the cumulative discounted reward is categorical, while Bellemare et al. (2019) established the convergence for policy evaluation using linear models from an algorithmic perspective. Both results were developed under the on-policy setting. In contrast, we studied the off-policy setting, where there is a distributional mis-match between the data collecting process and the target one, and consider the non-parametric models (i.e., neural network models) for estimating the distribution of cumulative discounted reward, which is thus more challenging and general than those previous analysis. Indeed, a recent comparative analysis (Lyle et al., 2019) showed that the improvement of distributional RL over the standard expected RL methods may come from the nonlinear function approximations, which also justifies the importance of our work.

As discussed before, our approach is closely related to the recently proposed Wasserstein-GANs (Arjovsky et al., 2017; Gulrajani et al., 2017*a*), which are powerful algorithms in

learning the generative model especially for high-dimensional random vectors. Theoretical development has also been made towards understanding their appealing empirical performance. See some recent results in Biau et al. (2021); Chen et al. (2020) and Chen et al. (2021) in the i.i.d. setting, and (Haas and Richter, 2020) in the time series analysis. Our distributional off-policy evaluation problem here is different from the standard learning task using Wasserstein GANs because the data from the target distribution are not directly observed in the infinite-horizon MDP. We rely on the distributional Bellman equation and formulate the estimation problem as a minimax two-play game as it can be roughly interpreted as some conditional moment model. Moreover, since we consider an estimation problem in a Markov process, temporal dependence among data is another challenge for statistical analysis. We show that our Wasserstein GAN-based estimator converges to the truth as long as either N or T approaches infinity under some technical conditions. Different from the analysis in Haas and Richter (2020), we do not require data come from the stationary distribution, which may provide a less restrictive application of our theoretical results.

Lastly, we review the literature of off-policy evaluation methods for estimating the value of a target policy, which has been extensively studied in the literature of RL. We mainly focused on the model-free methods. Most existing model-free off-policy evaluation (OPE) approaches can be divided into three categories. The first category is called direct methods (e.g, Ernst et al. (2005); Antos et al. (2008); Farahmand et al. (2016); Le et al. (2019); Shi et al. (2020) among others), which essentially make use of Bellman equation (e.g., Bertsekas (1995); Sutton and Barto (2018)) to estimate the Q-function. The second type of approaches is motivated by importance sampling (Precup, 2000; Xie et al., 2018) in the finite horizon setting, or recently proposed marginal importance sampling for the infinite horizon setting such as (Liu et al., 2018; Nachum et al., 2019; Xie et al., 2019; Uehara et al., 2020; Zhang, Dai, Li and Schuurmans, 2020; Zhang, Liu and Whiteson, 2020). The last type

of approaches combines direct methods and (marginal) importance sampling to construct the so-called doubly robust estimators (e.g., Jiang and Li (2016); Thomas and Brunskill (2016); Kallus and Uehara (2019); Tang et al. (2019); Shi et al. (2021) among many others). Our proposal is clearly different from aforementioned methods as we focus on estimating the joint distribution of multi-variate cumulative discounted reward, which is thus more challenging than estimating the value.

The rest of the paper is organized as follows. In Section 2, we present the framework of a time-homogeneous MDP and some necessary notations. Several motivating examples to illustrate the importance of studying the joint distribution of a multi-variate cumulative discounted reward are also introduced. In Section 3, we describe the distributional off-policy evaluation problem and our proposal using Wasserstein GANs. Statistical analysis is then provided in Section 4. Section 5 presents several numerical results related to our method and demonstrate its superiority. We conclude our paper in Section 6. Finally, all the proofs and additional simulation results can be found in the appendix.

2 Preliminaries and Motivating Examples

2.1 Problem Formulation and Notations

Our problem is driven by the empirical success of the recently developed distributional reinforcement learning (Bellemare et al., 2017; Dabney, Rowland, Bellemare and Munos, 2018). We aim to develop a method to simultaneously estimating the joint distributions of multivariate cumulative discounted reward (given any initial state-action) for a fixed policy using the batch data.

Consider a single trajectory $\{(S_t, A_t, R_t)\}_{t\geq 0}$ where (S_t, A_t, R_t) denotes the state-action-reward triplet collected at time t. We use $\mathcal{S} \subseteq \mathbb{R}^p$ and $\mathcal{A}$ to denote the state and action spaces, respectively. We assume $\mathcal{A}$ and $\mathcal{S}$ are discrete and finite to simplify our analysis.

The immediate reward R_t is a m-dimensional vector, i.e., $R_t \in \mathbb{R}^m$. For simplicity, we assume R_t are uniformly bounded by $R_{\max}$, i.e., $|R_{t,i}| \leq R_{\max}$ for every $1 \leq i \leq m$ and $t \geq 0$. A policy defines the agent's way of choosing the action at each decision time t. Throughout this paper, we are focused on the class of all stationary policies. Specifically, at each time point t, a stationary policy π maps the current state value into a probability mass function over the action space. For example, $\pi(a|s)$ denotes the probability of choosing action $a \in \mathcal{A}$ given the state value $s \in \mathcal{S}$. The goal of off-policy evaluation is to use the batch data generated by some possibly unknown policy to estimate the performance of any target policy π.

In the most existing literature, the target of interest for evaluating the performance of some policy π is the expected cumulative discounted reward (also called value). We first define a value function of a target policy π as

$$V^{\pi}(s) = \sum_{t=0}^{+\infty} \gamma^t \mathbb{E}^{\pi}(R_t | S_0 = s), \tag{1}$$

given an initial state s, where $\mathbb{E}^{\pi}$ denotes the expectation where all actions are selected according to π, and $0 \leq \gamma < 1$ refers to a discounted factor that balances the trade-off between immediate and future rewards. Based on the value function, one can define a value of a policy as

$$\mathcal{V}(\pi) = (1-\gamma) \sum_{s \in \mathcal{S}} V^{\pi}(s) \nu(s), \tag{2}$$

where ν denotes some reference distribution function over $\mathcal{S}$. The reference distribution ν is typically assumed known in the literature. Note that both $V^{\pi}(s)$ for every $s \in \mathcal{S}$ and $\mathcal{V}(\pi)$ are vector-valued function in our setting. The following two assumption serve as the foundations of the most existing OPE algorithms:

Assumption 1. *There exists a transition probability function p such that*

$$\Pr(S_{t+1} = s' | A_t = a, S_t = s, \{S_j, A_j, R_j\}_{0 \le j < t}) = p(s'|a, s),$$

for any $t \ge 1$, $a \in \mathcal{A}$ and $s, s' \in \mathcal{S}$.

Assumption 2. *The multivariate immediate reward R_t is a known vector-valued function of S_t, A_t and S_{t+1}, i.e, $R_t = \widetilde{\mathcal{R}}(S_t, A_t, S_{t+1})$, for any $t \ge 0$, where $\widetilde{\mathcal{R}} : \mathbb{R}^{2p+1} \to \mathbb{R}^m$. We further assume the range of $\widetilde{\mathcal{R}}$ is $[-R_{\max}, R_{\max}]^m$ for simplicity.*

By making Assumption 1 and 2, Bellman equation can be used to estimate $V^\pi(s)$ or $\mathcal{V}(\pi)$, for example via estimating the state-action value function (e.g.,Bertsekas (1995); Sutton and Barto (2018)) defined as

$$Q^\pi(s, a) = \mathbb{E}^\pi \left[\sum_{t=0}^{\infty} \gamma^t R_t \,|\, S_0 = s, A_0 = a \right], \tag{3}$$

for every $s \in \mathcal{S}$ and $a \in \mathcal{A}$. The state-action value function is also known as the Q-function. As we can see from (1) and (2), $V^\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) Q^\pi(s, a)$ and $\mathcal{V}(\pi) = \sum_{a \in \mathcal{A}, s \in \mathcal{S}} \pi(a|s) Q^\pi(s, a) \nu(s)$. In order to estimate Q^π, by Assumptions 1 and 2, one can obtain the following Bellman equation

$$Q^\pi(s, a) = \mathbb{E} \left[R_t + \gamma \sum_{a \in \mathcal{A}} \pi(a' \,|\, S_{t+1}) Q^\pi(S_{t+1}, a') \,|\, S_t = s, A_t = a \right], \tag{4}$$

for every $t \ge 0$, $s \in \mathcal{S}$ and $a \in \mathcal{A}$. Clearly (4) forms a conditional moment model, based on which Q^π can be estimated by many statistical methods such as generalized method of moments (Hansen, 1982) or nonparametric instrumental variable regression (Newey and Powell, 2003; Blundell et al., 2007; Chen and Christensen, 2018). We remark that many existing approaches for estimating Q^π can be naturally extended from the setting of a single reward to a multivariate one as we consider here.

In this paper, our target is much broader than the aforementioned goal. Instead of evaluating a policy by its value, we aim to learn the joint distributional behavior of the multi-variate cumulative discounted reward given any initial state-action pair. Define a m-dimensional random vector $Y = (Y_1, \cdots, Y_m) = \sum_{t=0}^{\infty} \gamma^t R_t$. Given a target policy π, our goal is to estimate the joint conditional distribution of Y for any initial state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$. In the latter section (Section 2.2), we provide several motivating examples to further illustrate the essential of this task.

Before that, we introduce several notations for the development of our proposal. As we study the batch setting, assume that batch data consisting of N trajectories are pre-collected and given, which correspond to N independent and identically distributed copies of $\{(S_t, A_t, R_t)\}_{t\geq 0}$. For each $i = 1, \cdots, N$, data collected from the ith trajectory can be summarized as $(S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1})\}_{0\leq t<T}$, where T denotes the termination time for each trajectory. We make the following assumption on our data generating mechanism.

Assumption 3. *The observed data $\mathcal{D}_N = \{S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1}\}_{1\leq i\leq N, 0\leq t\leq T-1}$ are generated by a fixed stationary policy b.*

In the literature, b is often called behavior policy. Next, for any $t \geq 0$, define $p_t^b(s, a)$ as the marginal probability mass function of (S_t, A_t) at $(s, a) \in \mathcal{S} \times \mathcal{A}$ under the behavior policy b. Then we can define the average visitation probability mass function under the behavior policy as

$$\bar{d}_T^b(s, a) = \frac{1}{T} \sum_{t=0}^{T-1} p_t^b(s, a)$$

for every $(s, a) \in \mathcal{S} \times \mathcal{A}$. For the notational simplicity, we use $\mathbb{E}$ for $\mathbb{E}^b$ when there is no confusion. Under Assumption 1 and 3, the process $\{(S_t, A_t)\}_{t\geq 0}$ forms a time-homogeneous Markov chain. For a function h, we define its Lipschitz norm as $\|h\|_{\text{Lip}} \triangleq \sup_{x\neq y} \frac{|h(x)-h(y)|}{\|x-y\|_2}$. Based on this notation, we define 1-Wasserstein metric between two distributions $\mathbb{P}_1$ and $\mathbb{P}_2$ over a compact space $\mathcal{T}$ as $W_1(\mathbb{P}_1, \mathbb{P}_2) = \sup_{h:\|h\|_{\text{Lip}}\leq 1} \int_{\mathcal{T}} h d\mathbb{P}_1 - \int_{\mathcal{T}} h d\mathbb{P}_2$.

By convention, we may also write as $W_1(U, X)$ the 1-Wasserstein metric between any

two random vectors U and X. We use uniform$(0,1)^m$ to denote the multi-variate uniform distribution over $[0,1]^m$, where each coordinate is independent from others. Finally, for generic sequences $\{\varpi(N)\}_{N\geq 1}$ and $\{\theta(N)\}_{N\geq 1}$, the notation $\varpi(N) \gtrsim \theta(N)$ (resp. $\varpi(N) \lesssim \theta(N)$) means that there exists a sufficiently large constant (resp. small) constant $c_1 > 0$ (resp. $c_2 > 0$) such that $\varpi(N) \geq c_1\theta(N)$ (resp. $\varpi(N) \leq c_2\theta(N)$). We use $\varpi(N) \asymp \theta(N)$ when $\varpi(N) \gtrsim \theta(N)$ and $\varpi(N) \lesssim \theta(N)$.

2.2 Motivating Examples

In this subsection, we present several motivating examples to illustrate the importance of estimating the joint distribution of Y given any target policy π.

The first example is related to the *pure exploration*. In recent years, there is a surging interest in studying the reward-free exploration of RL (Hazan et al., 2019; Du et al., 2019; Misra et al., 2020; Jin et al., 2020). The main idea is to design a suitable policy to collect effective samples of state-action transitions for space exploration without a pre-specified reward function. After the exploration stage, once the reward function is given, a batch RL algorithm can be efficiently implemented to learn an optimal policy. To achieve this goal, one approach is to find a policy that maximizes the coverage of the entire feature space. Define a q-dimensional state-action feature as $\psi(s,a) \in \mathbb{R}^q$ for every $(s,a) \in \mathcal{S} \times \mathcal{A}$. Similar to that in (Zhang, Koppel, Bedi, Szepesvari and Wang, 2020), one can find a policy that

$$\max_{\pi} \mathbb{E}^{\pi} \left[\lambda_{\min} \left(\sum_{t=0}^{+\infty} \gamma^t \psi(S_t, A_t) \psi(S_t, A_t)^{\top} \right) \right], \tag{5}$$

where $\lambda_{\min}(\bullet)$ denotes the minimum eigenvalue of a square matrix. To solve problem (5), one possible approach is to first estimate the joint distribution of $\sum_{t=0}^{+\infty} \gamma^t \phi(S_t, A_t)\phi(S_t, A_t)^{\top}$ given any policy π, which therefore can be regarded as a special case of our problem described in the previous subsection by letting $R_t = \text{vector}(\phi(S_t, A_t)\phi(S_t, A_t)^{\top})$, where vector$(\bullet)$ denotes the vectorization. Once the distribution is learned, steps for the policy

optimization with respect to (5) could be implemented.

The second example is related to *risk constraint/sensitive reinforcement learning* (e.g., Prashanth and Ghavamzadeh (2013); Shen et al. (2014); Chow et al. (2017); Zhong et al. (2020)), which plays an important role in the domain of safe RL (Garcıa and Fernández, 2015). In general, such problems can be formulated as

$$\begin{aligned} &\max_{\pi} \ \ h_1^{\pi}(Y_1) \\ &\text{subject to} \ \ h_i^{\pi}(Y_i) \le c_i, \ \ \text{for } 2 \le i \le m, \end{aligned} \tag{6}$$

where, for $1 \le i \le m$, h_i^{π} are some known (utility or risk) functions related to π and c_i are some constants. In particular, (Chow et al., 2017) considered h_1^{π} as an expectation operator $\mathbb{E}^{\pi}$ of Y_1 and h_2^{π} as the conditional-value-at-risk (CVaR) of Y_2 (e.g., (Ben-Tal and Teboulle, 1986; Rockafellar et al., 2000)). Therefore they consider a two dimensional reward function at each decision point t. They also pointed out several alternatives for h_2^{π} such as the value-at-risk (i.e., quantile) or the chance-constraint, e.g.,

$$\Pr^{\pi}(Y_2 \ge \widetilde{c}_1) \ge \widetilde{c}_2,$$

for some constants $\widetilde{c}_1$ and $\widetilde{c}_2 \in [0, 1]$, where $\Pr^{\pi}$ denotes the probability distribution under the target policy π. The state-of-art method for solving (6) is to apply some Lagrangian methods. In order to have more flexibility of choosing different forms of h_i^{π} in (6), it is necessary to learn the joint distribution of Y given any policy π so as the Lagrangian method can be easily applied.

Our last motivating example is related to the multi-objective reinforcement learning (MORL) (Roijers et al., 2013). One typical approach for solving MORL problems is to scalarize the multi-objective value $\mathcal{V}(\pi)$ by some either linear or nonlinear function f indexed by θ, i.e., $\widetilde{\mathcal{V}}_{\theta}(\pi) = f(\mathcal{V}(\pi), \theta)$. Based on this scalarization, one may aim to compute the

so-called *Pareto front* set of policies. See Section 4.2.2 of (Roijers et al., 2013) for more details. While one can compute $\widetilde{\mathcal{V}}_\theta(\pi)$ via estimating each coordinate of $\mathcal{V}(\pi)$ separately, it may be helpful to compute the joint distribution of Y for capturing relationships among different coordinates of rewards so as to improve the efficiency. Furthermore, as discussed by (Roijers et al., 2013), it is also meaningful to consider the *expectation after the scalarized return* (ESR), i.e., $\widetilde{\mathcal{V}}_\theta(\pi) = \mathbb{E}^\pi\left[f(Y,\theta)\right]$, which may be more appropriate in some practical applications (e.g., Lizotte et al. (2010)). In this case, it may be necessary to study the joint distribution of Y when some general function f is considered.

As seen from these three examples, it is vital to investigate how to estimate the joint distribution of Y given a target policy π using the batch data. Finally we remark that in most aforementioned examples, the existing state-of-art methods such as Implicit Quantile Networks (IQN) cannot be directly applied because when the reward is multi-variate, the Wasserstein-typed distance cannot be directly computed via quantiles. This also motivates our proposal below.

3 Distributional Off-Policy Evaluation via Wasserstein GANs

In this section, we propose a distributional off-policy evaluation method to estimate the joint distribution of Y given any initial state-action pair for a fixed policy using the batch data. The proposed method is based on the distributional Bellman equation introduced below.

3.1 Distributional Bellman Equation

Recall that $Y = \sum_{t=0}^{+\infty} \gamma^t R_t$, whose support is $[-\frac{R_{\max}}{1-\gamma}, \frac{R_{\max}}{1-\gamma}]^m$ by Assumption 2. Given a target policy π, our goal is to estimate the distribution of Y given any initial state $s \in \mathcal{S}$

and initial action $a \in \mathcal{A}$ under this policy. We use $\mathcal{U}$ to denote the product space of Y, states S and actions A, i.e., $\mathcal{U} = [-\frac{R_{\max}}{1-\gamma}, \frac{R_{\max}}{1-\gamma}]^m \times \mathcal{S} \times \mathcal{A}$. For any $y \in \mathbb{R}^m$, define the conditional distribution of Y under the policy π given a state-action pair (s, a) as

$$\Pr^{\pi}(Y \leq y \,|\, S_0 = s, A_0 = a) \triangleq F_Y^{\pi}(y, s, a),$$

where the subscript in F_Y^{π} indicates that actions are selected according to π. In order to estimate F_Y^{π}, we have the following distributional Bellman equation.

Lemma 1. *Under Assumptions 1-2, the conditional distribution function F_Y^{π} satisfies that for any $t \geq 0$, $y \in \mathbb{R}^m$, $s \in \mathcal{S}$ and $a \in \mathcal{A}$,*

$$F_Y^{\pi}(y, s, a) = \mathbb{E}\left[\sum_{a' \in \mathcal{A}} \pi(a'|S_{t+1}) F_Y^{\pi}(\frac{y - R_t}{\gamma}, S_{t+1}, a') \,|\, S_t = s, A_t = a\right]. \tag{7}$$

Also see Jaquette (1973); Chung and Sobel (1987); Bellemare et al. (2017) for more details. To understand Lemma 1, let $\mathbb{P}(\mathbb{R}^m)$ be the space of all probability distributions over $\mathbb{R}^m$. Denote $\mathcal{L}(X)$ as the probability distribution of any random vector X. Define the distributional Bellman operator as $\mathcal{T}^{\pi} : \mathbb{P}(\mathbb{R}^m)^{\mathcal{S}\times\mathcal{A}} \to \mathbb{P}(\mathbb{R}^m)^{\mathcal{S}\times\mathcal{A}}$ such that

$$\mathcal{T}^{\pi}\Upsilon(s, a) = \sum_{a' \in \mathcal{A}, s' \in \mathcal{S}} \mathcal{L}(\widetilde{R}(s, a, s')) \circ \mathcal{L}(\gamma Y(s', a'))\pi(a' \,|\, s')p(s' \,|\, s, a), \tag{8}$$

for any $(s, a) \in \mathcal{S} \times \mathcal{A}$, where $\circ$ denotes the convolution of two distributions, $Y(s, a)$ refers to some random vector indexed by (s, a) with probability distribution $\Upsilon(s, a)$, and $\widetilde{R}(s, a, s')$ is the deterministic function defined in Assumption 2. Then Lemma 1 implies that $\mathcal{T}^{\pi} F_Y^{\pi} = F_Y^{\pi}$, which can be seen as an equivalence of (7). We remark that if there is an external noise on the reward function, the distributional Bellman equation (7) is still valid. In addition, Equation (7) in Lemma 1 can also be understood as a conditional moment model similar to (4) for every $y \in \mathbb{R}^m$. One may consider to use for example two-stage least squares regression

to estimate the distribution F_Y^π similar to that in the Q-function estimation. However, the difference between (7) and the standard conditional moment model lies in that Model (7) is indexed by a vector y that takes values in an interval. Therefore how to effectively construct a two-stage least squares etimator remains unknown. In the following, we propose a different approach by making use of the recently developed generative adversarial networks (Goodfellow et al., 2014, GANs).

Our proposal is motivated by the contraction property of the distributional Bellman operator. Consider $\Upsilon_1, \Upsilon_2 \in \mathbb{P}(\mathbb{R}^m)^{\mathcal{S}\times\mathcal{A}}$. Define a supreme 1-Wasserstein metric as

$$\bar{d}_1(\Upsilon_1, \Upsilon_2) = \sup_{s\in\mathcal{S}, a\in\mathcal{A}} W_1(\Upsilon_1(s,a), \Upsilon_2(s,a)), \tag{9}$$

We have the following lemma to show the distributional Bellman operator $\mathcal{T}^\pi$ is γ-contractive under $\bar{d}_1$.

Lemma 2. *Under Assumptions 1-2, $\bar{d}_1$ is γ-contractive, i.e., for any $\Upsilon_1, \Upsilon_2 \in \mathbb{P}(\mathbb{R}^m)^{\mathcal{S}\times\mathcal{A}}$,*

$$\bar{d}_1(\mathcal{T}^\pi\Upsilon_1, \mathcal{T}^\pi\Upsilon_2) \le \gamma\bar{d}_1(\Upsilon_1, \Upsilon_2). \tag{10}$$

Indeed one can show that if a probability metric (or divergence) is (γ, r)-perfect for some $r > 0$, then the distributional Bellman operator is also γ-contractive under the supreme of this metric over $\mathcal{S}\times\mathcal{A}$. See the definition of the perfect metric in Zolotarev (1976) with examples such as Wasserstein and Crámer distance (Rowland et al., 2018). Based on Lemma 2, we propose a Wasserstein GAN-based distributional off-policy evaluation method to estimate F_Y^π. In the following, we present one key lemma as the foundation of our proposed method.

Lemma 3. *For any function h defined over $\mathbb{R}^m \times \mathcal{S} \times \mathcal{A}$, the following equation holds.*

$$\begin{aligned} &\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}^{\pi}\left\{h(\sum_{t'\geq t}^{+\infty}\gamma^{t'-t}R_{t'},S_t,A_t)\,|\,S_t,A_t\right\}\right] \qquad (11)\\ =&\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}^{\pi}\left\{\sum_{a'\in\mathcal{A}}\pi(a'\,|\,S_{t+1})h(\gamma\sum_{t'\geq t+1}^{+\infty}\gamma^{t'-t-1}R_{t'}+R_t,S_t,A_t)\,|\,A_{t+1}=a',S_{t+1},S_t,A_t\right\}\right]. \end{aligned}$$

Due to the time-homogeneity of our Markov model, the distribution of $\sum_{t'\geq t}^{+\infty}\gamma^{t'-t}R_{t'}$ is the same as that of $\sum_{t'\geq t+1}^{+\infty}\gamma^{t'-t-1}R_{t'}$. Motivated by Lemma 3, we aim to learn a generator $\mathbb{G}^{\pi}$ that takes any state-action pair (s,a) and a random vector $\mathbf{Z}\sim\text{uniform}(0,1)^m$ as the input, and outputs a random sample $\widetilde{Y}$ that follows the target conditional distribution $F_Y^{\pi}(\bullet,s,a)$. Such a procedure is also inspired by the conditional GANs introduced by (Mirza and Osindero, 2014). In the typical conditional GANs such as (Arjovsky et al., 2017), the generator to produce $\widetilde{Y}$ is usually trained by minimizing some divergence between the targeted conditional distribution of Y and $\widetilde{Y}$ given every pair (s,a) if random samples of Y are observed. Alternatively, a generator $\mathbb{G}^{\pi}(\mathbf{Z},S,A)$ can be trained so that the joint distribution of (Y,S,A) is the same as $(\mathbb{G}^{\pi}(\mathbf{Z},S,A),S,A)$. However, different from the conventional probability distribution estimating task, Y is not observed in our problem. Nevertheless, distributional Bellman equations (7) with Lemma 3 inspire us to estimate the generator $\mathbb{G}^{\pi}$ by solving the following min-max optimization problem, which has the similar spirit of Wasserstein-GANs in (Arjovsky et al., 2017).

$$\begin{aligned} \underset{\mathbb{G}}{\text{minimize}}\sup_{h:\|h\|_{\text{Lip}}\leq 1}\Bigg\{&\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\sum_{a\in\mathcal{A}}\pi(a|S_{t+1})h(\gamma\mathbb{G}(\mathbf{Z_t},S_{t+1},a)+R_t,S_t,A_t)\right]\\ &-\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}h(\mathbb{G}(\widetilde{\mathbf{Z}}_\mathbf{t},S_t,A_t),S_t,A_t)\right]\Bigg\}, \qquad (12) \end{aligned}$$

where $\{\mathbf{Z_t}\}_{0\leq t\leq(T-1)}$ and $\{\widetilde{\mathbf{Z}}_\mathbf{t}\}_{0\leq t\leq(T-1)}$ are set to independently follow $\text{uniform}(0,1)^m$, and h, which is called discriminator, is a function mapping from $\mathbb{R}^{p+m+1}$ to $\mathbb{R}$.

In practice, as in the conventional GANs, we restrict h in Problem (12) to some set

of neural networks $\mathcal{R}_D$ defined in Section 3.2, which can provide a tractable computation. More importantly, by the powerful deep neural networks, we are able to handle with the potential high-dimensional state-action space $\mathcal{S} \times \mathcal{A}$ (i.e., the number of states and actions is huge) and reward function (i.e., m is large). We also use some set of neural networks $\mathcal{R}_G$ to model the generator $\mathbb{G}^\pi$ in order to capture the potentially complex distribution F_Y^π. Below, we explain the structure of neural networks considered in this paper.

3.2 ReLU Neural networks

In this subsection, we specify the classes $\mathcal{R}_D$ and $\mathcal{R}_G$ of neural networks in details. We consider similar configurations of neural networks as those in (Schmidt-Hieber, 2020) and (Haas and Richter, 2020), and thus follow their notations. Define a rectified linear unit (ReLU) activation function as $\sigma(x) = \max\{x, 0\}$ for $x \in \mathbb{R}$. For two vectors $v, x \in \mathbb{R}^{\widetilde{p}}$, where $\widetilde{p}$ is some generic integer, $\sigma_v(x) = \sigma(x - v)$ refers to coordinate-wisely applying $\sigma(\bullet)$ on the vector $x - v$. In neural network literature, v is usually called bias vector instead of intercepts in statistics. Let an integer L be the number of hidden layers or depths of a neural network with a width vector denoted by $\mathbf{p} = (p_0, ..., p_{L+1})$. In particular, p_0 is the width (the number of neurons) of the input layer and p_{L+1} is the width of the output layer. A neural network with an architecture $(L, \mathbf{p})$ is defined as a function $h : \mathbb{R}^{p_0} \to \mathbb{R}^{p_{L+1}}$ such that

$$h(x) = W^{(L)}\sigma_{v^{(L)}}W^{(L-1)} \dots W^{(1)}\sigma_{v^{(1)}}W^{(0)}x, \tag{13}$$

where $W^{(l)} \in \mathbb{R}^{p_{l+1} \times p_l}$ are the weight matrices for $l = 0, ..., L$, and $v^{(l)} \in \mathbb{R}^{p_l}$, $l = 1, ..., L$ are the bias vectors associated to the network. We denote a generic class of neural networks with the number of hidden layers L and the width $\mathbf{p}$ by

$$\mathcal{R}(L, \mathbf{p}) = \Big\{ h : \mathbb{R}^{p_0} \to \mathbb{R}^{p_{L+1}} \mid h \text{ has the form (13)} \Big\}.$$

Empirical findings show that in many applications sparse neural networks perform as well as the dense one but enjoy more computational advantage (Frankle and Carbin, 2018). Therefore, motivated by (Schmidt-Hieber, 2020), we also adopt a set of sparse networks bounded by some threshold $F > 0$ for modeling generators $\mathbb{G}$ and discriminators h in (12), which is defined as

$$\mathcal{R}(L, \mathbf{p}, s, F) := \Big\{ h \in \mathcal{R}(L, \mathbf{p}) \;\Big|\; \max_{j=0,\ldots,L} \|W^{(j)}\|_\infty \vee \max_{j=1,\ldots,L} \|v^{(j)}\|_\infty \le 1, \\ \sum_{j=0}^{L} \|W^{(j)}\|_0 + \sum_{j=1}^{L} \|v^{(j)}\|_0 \le s, \text{ and } \|h\|_\infty \le F \Big\}.$$

We fix F throughout this paper, and hence may write $\mathcal{R}(L, \mathbf{p}, s, F)$ by $\mathcal{R}(L, \mathbf{p}, s)$ for brevity. In our problem (12), we use $\mathcal{R}_G = \mathcal{R}(L_{\mathbb{G}}, \mathbf{p}_{\mathbb{G}}, s_{\mathbb{G}})$ to approximate the generator and $\mathcal{R}_h = \mathcal{R}(L_h, \mathbf{p}_h, s_h)$ for the discriminator h. Clearly, the dimensions of the input layer in both neural networks are $p_0 = p + m + 1$, while the dimension of the output layer in $\mathcal{R}_G$ is m and 1 in $\mathcal{R}_D$ respectively.

3.3 Estimation

In this subsection, we discuss our estimation procedure for learning $\mathbb{G}^\pi(\bullet, s, a)$ using the batch data $\mathcal{D}_N = \{(S_{i,t}, A_{i,t}, R_{i,t}, S_{i,t+1})\}_{0 \le t \le (T-1), 1 \le i \le N}$ for the conditional distribution of Y given $s \in \mathcal{S}$ and $a \in \mathcal{A}$ under the target policy π. Specifically, we use the empirical average to estimate the objective function in (12) and solve the following optimization problem to estimate the generator $\mathbb{G}^\pi$.

$$\underset{\mathbb{G} \in \mathcal{R}(L_{\mathbb{G}}, \mathbf{p}_{\mathbb{G}}, s_{\mathbb{G}})}{\text{minimize}} \sup_{h \in \mathcal{R}(L_h, \mathbf{p}_h, s_h), \|h\|_{\text{Lip}} \le 1} \Bigg\{ \frac{1}{NT} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \sum_{a \in \mathcal{A}} \pi(a | S_{i,t+1}) h(\gamma \mathbb{G}(\mathbf{Z_{i,t}}, S_{i,t+1}, a) + R_{i,t}, S_{i,t}, A_{i,t}) \\ - \frac{1}{NT} \sum_{i=1}^{N} \sum_{t=0}^{T-1} h(\mathbb{G}(\widetilde{\mathbf{Z}}_{\mathbf{i,t}}, S_{i,t}, A_{i,t}), S_{i,t}, A_{i,t}) \Bigg\}, \tag{14}$$

where $\{\mathbf{Z_{i,t}}\}_{1\le i\le N;0\le t\le(T-1)}$ and $\{\widetilde{\mathbf{Z}}_{\mathbf{i,t}}\}_{1\le i\le N;0\le t\le(T-1)}$ are two set of independent samples both generated from uniform$(0,1)^m$, and $(L_{\mathbb{G}},\mathbf{p}_{\mathbb{G}},s_{\mathbb{G}})$ and $(L_h,\mathbf{p}_h,s_h)$ are neutral network configurations of the generator $\mathbb{G}$ and the discriminator h respectively. See the detailed definition in Section 3.2. The optimal solution of the optimization (14) always exists because the inner maximization problem (after taking supreme) is Lipschitz with respect to $\mathbb{G}$ and furthermore $\mathbb{G}$ is Lipschitz with respect to all parameters in $\mathcal{R}(L_{\mathbb{G}},\mathbf{p}_{\mathbb{G}},s_{\mathbb{G}})$ and all parameters have bounded supports. Similarly, the inner maximization problem also has an optimal solution. Stochastic gradient methods, which are the state-of-the-art, can be implemented to solve the above optimization problem. Denote the minimizer of (14) as $\widehat{\mathbb{G}}^\pi$. Then we can use $\widehat{\mathbb{G}}^\pi(\bullet,s,a)$ to generate a sequence of pseduo samples $\{\widetilde{Y}_j^\pi(s,a)\}_{1\le j\le M}$ for some integer M to approximate the conditional distribution $F_Y^\pi(\bullet,s,a)$ for every (s,a). See Algorithm 1 in the appendix for more details. Finally we can compute many interesting statistics related to the target policy π. In the following, we provide several examples.

Example 1 (Value of a target policy)**.** *If the reference distribution ν is given, we can estimate $\mathcal{V}(\pi)$ by $\widehat{\mathcal{V}}_1(\pi)=\frac{1}{M}\sum_{j=1}^M\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\pi(a\,|\,s)\widetilde{Y}_m^\pi(s,a)\nu(s)$. If we do not know ν but can obtain random samples from ν, we can generate a sequence of initial states $\{\widetilde{S}_{0,k}\}_{1\le k\le K}$ and use the following quantity to estimate $\mathcal{V}(\pi)$. $\widehat{\mathcal{V}}_2(\pi)=\frac{1}{MK}\sum_{j=1}^M\sum_{k=1}^K\sum_{a\in\mathcal{A}}\pi(a\,|\,\widetilde{S}_{0,k})\widetilde{Y}_j^\pi(\widetilde{S}_{0,k},a)$.*

Example 2 (Probability of some events on the multivariate discounted cumulative reward)**.** *If some chance constraints are imposed in the policy optimization, one often needs to estimate the probability of certain events related to a multivariate discounted cumulative reward such as $\Pr^\pi(Y\le y)$ for some pre-specified constant y. Then given the psuedo samples, we can estimate this probability by $\widehat{P}^\pi(Y\le y)=\frac{1}{M}\sum_{j=1}^M\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\pi(a\,|\,s)\nu(s)\mathbb{I}\left(\widetilde{Y}_j^\pi(s,a)\le y\right)$, where $\mathbb{I}(\bullet)$ denotes the indicator function.*

Example 3 (Covariance matrix of the multivariate discounted cumulative reward)**.** *We can estimate the covariance of Y under the target policy π, i.e., $Var^\pi(Y)$, by $\widehat{Var}^\pi(Y)=\frac{1}{M}\sum_{j=1}^M\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\pi(a\,|\,s)\nu(s)\left(\widetilde{Y}_j^\pi(s,a)-\widehat{\mathcal{V}}(\pi)\right)\left(\widetilde{Y}_j^\pi(s,a)-\widehat{\mathcal{V}}(\pi)\right)^\top$ where $\widehat{\mathcal{V}}(\pi)$ can be*

either $\widehat{\mathcal{V}}_1(\pi)$ *or* $\widehat{\mathcal{V}}_2(\pi)$ *given in the Example 1.*

In all our examples, if $\widehat{\mathbb{G}}^\pi$ is consistent to F_Y^π and M (or K) diverges to infinity, $\widehat{\mathcal{V}}_1(\pi)$, $\widehat{\mathcal{V}}_2(\pi)$, $\widehat{P}^\pi(Y \le y)$ and $\widehat{\mathrm{Var}}^\pi(Y)$ are all consistent.

4 Statistical Analysis

In this section, we study some statistical properties related to our method. Consider a class of functions $h : \mathcal{T} \subseteq \mathbb{R}^r \to \mathbb{R}$ with Hölder coefficient $\beta \ge 1$ as

$$C^\beta(\mathcal{T}, K) \triangleq \left\{ h : \mathcal{T} \to \mathbb{R} \mid \sup_{0 \le \|\alpha\|_1 \le \lfloor\beta\rfloor} \|\partial^\alpha h\|_\infty + \sup_{\alpha : \|\alpha\|_1 = \lfloor\beta\rfloor} \sup_{x \ne y} \frac{|\partial^\alpha h(x) - \partial^\alpha h(y)|}{\|x - y\|_2^{\beta - \lfloor\beta\rfloor}} \le K \right\}. \quad (15)$$

where $\lfloor\beta\rfloor$ denotes the integer no larger than β for any $\beta > 0$, $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_r)$ and

$$\partial^\alpha h(x) = \frac{\partial^\alpha h(x)}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_d^{\alpha_r}}.$$

Due to the challenge of the distributional off-policy evaluation, especially when the dimension of $\mathcal{S}$ and m are large, we impose some structure assumptions on the underlying true distribution to simplify our analysis. In particular, suggested by (Haas and Richter, 2020), we assume the distribution of Y under the target policy π given any state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$ lies in some $d_{\mathbb{G}}$-dimensional subspace with $d_{\mathbb{G}} < p + m + 1$. Based on this consideration, we use the following functional class, which we assume the generator $\mathbb{G}^\pi$ belongs to.

Definition 4.1. *Let* $\mathcal{G}(d_{\mathbb{G}}, \beta, K)$ *with* $\beta \ge 1$ *be the class of all measurable vector-valued functions* $\mathbb{G} : \mathbb{R}^{p+m+1} \to \mathbb{R}^m$ *such that any component* $\{\mathbb{G}_i\}_{1 \le i \le m}$ *depends on at most* $d_{\mathbb{G}}$ *arguments and lies in* $C^\beta(\mathcal{U}, K)$.

We make several technical assumptions below.

Assumption 4. *There exists a positive constant* $p_{\min}$ *such that* $\bar{d}_T^b(s,a) \geq p_{\min}$ *for every* $s \in \mathcal{S}$ *and* $a \in \mathcal{A}$.

Assumption 5. *The stochastic process* $\{S_t, A_t\}_{t\geq 0}$ *is geometrically ergodic, i.e, there exists some function* $\phi(s,a)$ *and a constant* $\rho \in (0,1)$ *such that for any* (s,a),

$$\|p_t^b(\cdot \mid (s,a)) - d^b(\cdot)\|_{TV} \leq \phi(s,a)\rho^t, \qquad \forall t \geq 0,$$

where d^b *is the behavior policy induced stationary distribution,* $\|\bullet\|_{TV}$ *denotes the total variation norm and* $\mathbb{E}[\phi(S_0,A_0)] \leq C$ *for some positive constant* C.

Assumption 4 is also called *coverage* assumption frequently used in the literature of RL such as (Precup, 2000; Antos et al., 2008; Kallus and Uehara, 2019) among many others. It essentially states that every state-action pair has some positive probability of being observed. Assumption 5 states that there exists a stationary distribution induced by the behavior policy and the Markov chain converges to this stationary distribution exponentially fast. This assumption is used to derive a sharp bound of the suprema of some empirical process in terms of both N and T. See Lemma 3.1 in the appendix. Define

$$\sup_{h:\|h\|_{\text{Lip}}\leq 1} \left\{ \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\sum_{a\in\mathcal{A}} \pi(a|S_{t+1}) h(\gamma \mathbb{G}(\mathbf{Z_t}, S_{t+1}, a) + R_t, S_t, A_t)\right] \right.$$
$$\left. -\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1} h(\mathbb{G}(\mathbf{Z_t}, S_t, A_t), S_t, A_t)\right] \right\} \triangleq \sup_{h:\|h\|_{\text{Lip}}\leq 1} \overline{W}(h,\mathbb{G}) \triangleq W^*(\mathbb{G}),$$

and

$$W_{NT}(\mathbb{G}) \triangleq \sup_{h\in\mathcal{R}(L_h,\mathbf{p}_h,s_h),\|h\|_{\text{Lip}}\leq 1} \left\{ \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=0}^{T-1}\sum_{a\in\mathcal{A}} \pi(a|S_{i,t+1}) h(\gamma\mathbb{G}(\mathbf{Z_{i,t}}, S_{i,t+1}, a) + R_{i,t}, S_{i,t}, A_{i,t}) \right.$$
$$\left. -\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=0}^{T-1} h(\mathbb{G}(\widetilde{\mathbf{Z}}_{\mathbf{i,t}}, S_{i,t}, A_{i,t}), S_{i,t}, A_{i,t}) \right\}.$$

In addition, let $W(\mathbb{G}) \triangleq \sup_{h\in\mathcal{R}(L_h,\mathbf{p}_h,s_h),\|h\|_{\text{Lip}}\leq 1} \overline{W}(h,\mathbb{G})$.

Assumption 6. *Suppose that* $\mathbb{G}^\pi \in \mathcal{G}(d_\mathbb{G}, \beta, K)$*, where the distribution of* $\mathbb{G}^\pi(\mathbf{Z}, s, a)$ *is the same as* $F_Y^\pi(\bullet, s, a)$ *for every* $s \in \mathcal{S}$ *and* $a \in \mathcal{A}$*. For any* $\mathbb{G} \in \mathcal{R}(L_\mathbb{G}, \mathbf{p}_\mathbb{G}, s_\mathbb{G})$*, there exists* $\widetilde{h} \in C^\theta(\mathcal{U}, \widetilde{K})$*, which depends on at most* d_h *arguments, for some universal positive constants* $\theta \geq 1$ *and* $\widetilde{K}$ *such that* $\widetilde{h} \in argmin_{h:\|h\|_{Lip}\leq 1} \overline{W}(h, \mathbb{G})$*. Denote the subset of* $C^\theta(\mathcal{U}, \widetilde{K})$ *in which functions depends on at most* d_h *arguments as* $\mathcal{H}(d_h, \theta, \widetilde{K})$*.*

Define $\xi_{NT} = \max\{(NT)^{-\frac{2\beta}{2\beta+d_\mathbb{G}}}, (NT)^{-\frac{2\theta}{2\theta+d_h}}\}$.

Assumption 7 (Neural Network)**.** *The following conditions hold:*

(a) $F \geq \max(K, \widetilde{K}, 1)$

(b) $\log_2(NT)\log_2(4\max(d_\mathbb{G}, \beta, d_h, \theta)) \leq L_\mathbb{G} \lesssim \log(NT)$

(c) $NT\xi_{NT} \leq \min\{\min_{1\leq i\leq L_\mathbb{G}} p_{\mathbb{G},i}, \min_{1\leq i\leq L_h} p_{h,i}\}$

(d) $s_\mathbb{G} \asymp (NT)\xi_{NT}\log(NT)$

(d) $L_h \leq L_\mathbb{G}, s_h \leq s_\mathbb{G}$

In Assumption 6, by assuming that $\mathbb{G}^\pi$ belongs to some Hölder class with smaller dimensions, we are able to simplify our analysis for controlling the approximation error to $\mathbb{G}^\pi$ using $\mathcal{R}_G$ and overcome the curse of dimensionality. The second condition in Assumption 6 is imposed to handle the approximation error caused by using the neural network model $\mathcal{R}_h$. Assumption 7 is mainly used to control the complexity of our neural networks, which is commonly imposed in the existing literature such as Schmidt-Hieber (2020). Denote $Y^\pi(s, a)$ as a random vector with distribution $F_Y^\pi(\bullet, s, a)$.

Theorem 1. *Under Assumptions 1- 4 and 6, there exists some universal constant* C *such*

that

$$\begin{aligned}
&\sup_{s\in\mathcal{S},a\in\mathcal{A}} W_1(Y^\pi(s,a),\widehat{\mathbb{G}}^\pi(\mathbf{Z},s,a)) \\
\leq&\frac{C}{p_{\min}(1-\gamma)}\sup_{h:\|h\|_{Lip}\leq 1}\left\{\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\sum_{a\in\mathcal{A}}\pi(a|S_{t+1})h(\gamma\widehat{\mathbb{G}}^\pi(\mathbf{Z_t},S_{t+1},a)+R_t,S_t,A_t)\right]\right. \\
&\left.-\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}h(\widehat{\mathbb{G}}^\pi(\mathbf{Z_t},S_t,A_t),S_t,A_t)\right]\right\}\lesssim W^*(\widehat{\mathbb{G}}^\pi)
\end{aligned} \tag{16}$$

Theorem 1 essentially implies that to bound the supreme 1-Wasserstein metric between $\widehat{\mathbb{G}}^\pi$ to the truth, it is enough to focus on bounding the distributional Bellman error of $\widehat{\mathbb{G}}^\pi$, i.e., $W^*(\widehat{\mathbb{G}}^\pi)$. The proof of Theorem 1 relies on the contraction property of distributional Bellman operator $\mathcal{T}^\pi$ defined in (8) under the supreme 1-Wasserstein metric. See Lemma 2 for more details. Next, we leverage the following theorem to decompose $W^*(\widehat{\mathbb{G}}^\pi)$ into estimation and approximation errors.

Theorem 2. *Under Assumptions 1- 3 and 6, we have*

$$\begin{aligned}
&W^*(\widehat{\mathbb{G}}^\pi) \\
\leq&2\sup_{h\in\mathcal{H}(d_h,\theta,\widetilde{K})}\inf_{\bar{h}\in\mathcal{R}(L_h,\boldsymbol{p}_h,s_h),\|\bar{h}\|_{Lip}\leq 1}\|h-\bar{h}\|_\infty \\
&+W(\widehat{\mathbb{G}}^\pi)-\inf_{\mathbb{G}\in\mathcal{R}(L_\mathbb{G},\mathbf{p}_\mathbb{G},s_\mathbb{G})}W(\mathbb{G})+\sqrt{m}(1+\gamma)\sup_{\widetilde{\mathbb{G}}\in\mathcal{G}(d_\mathbb{G},\beta,K)}\inf_{\mathbb{G}\in\mathcal{R}(L_\mathbb{G},\mathbf{p}_\mathbb{G},s_\mathbb{G})}\|\widetilde{\mathbb{G}}-\mathbb{G}\|_\infty \\
=&a_{1,NT}+e_{NT}+a_{2,NT},
\end{aligned} \tag{17}$$

where $a_{1,NT}\triangleq 2\sup_{h\in\mathcal{H}(d_h,\theta,\widetilde{K})}\inf_{\bar{h}\in\mathcal{R}(L_h,\boldsymbol{p}_h,s_h),\|\bar{h}\|_{Lip}\leq 1}\|h-\bar{h}\|_\infty$, $e_{NT}\triangleq W(\widehat{\mathbb{G}}^\pi)-\inf_{\mathbb{G}\in\mathcal{R}(L_\mathbb{G},\mathbf{p}_\mathbb{G},s_\mathbb{G})}W(\mathbb{G})$, *and* $a_{2,NT}\triangleq\sqrt{m}(1+\gamma)\sup_{\widetilde{\mathbb{G}}\in\mathcal{G}(d_\mathbb{G},\beta,K)}\inf_{\mathbb{G}\in\mathcal{R}(L_\mathbb{G},\mathbf{p}_\mathbb{G},s_\mathbb{G})}\|\widetilde{\mathbb{G}}-\mathbb{G}\|_\infty$

Theorem 2 decomposes $W^*(\widehat{\mathbb{G}}^\pi)$ into three parts. In particular, $a_{i,NT}$ for $i=1,2$ are approximation errors for the generator and discriminator respectively and e_{NT} is the estimation error. By using the result in the proof of Theorem 1 in (Schmidt-Hieber, 2020) and Lemma B.3 in (Haas and Richter, 2020), we have the following corollary for quantifying

these two approximation errors, where the proof is omitted for brevity.

Corollary 1. *Suppose Assumption 7 holds and let* $\widetilde{M} = NT\xi_{NT}$. *Then for sufficiently large* NT, *we have*

$$a_{1,NT} + a_{2,NT} \lesssim \sqrt{m}(1+\gamma)\left\{\frac{\widetilde{M}}{NT} + \widetilde{M}^{-\beta/d_{\mathbb{G}}} + \widetilde{M}^{-\theta/d_h}\right\} \lesssim \sqrt{m}(1+\gamma)\xi_{NT}^{1/2}. \tag{18}$$

Finally, we can derive the finite sample error bound for our proposed estimator by properly controlling the estimation error using the new concentration inequality developed in Lemma 3.1 in the appendix.

Theorem 3. *Under Assumptions 1-7, for sufficiently large* NT, *with probability at least* $1 - \frac{1}{NT}$,

$$\sup_{s\in\mathcal{S},a\in\mathcal{A}} W_1(Y^\pi(s,a), \widehat{\mathbb{G}}^\pi(\mathbf{Z},s,a)) \tag{19}$$

$$\lesssim \sqrt{m}\xi_{NT}^{1/2}\log^2(NT). \tag{20}$$

As seen from Theorem 3, we can show that the estimated generator will converge to the truth in the sense of the supreme 1-Wasserstein metric in terms of both N and T. This indicates that our estimator converges as long as either N or T approaches to infinity, which breaks the curse of horizon. Based on the property of supreme 1-Wasserstein metric, Theorem 3 also implies that the distribution and the first moment of $\widehat{\mathbb{G}}^\pi$ converge to the truth.

5 Experimental Results

In this section, we conduct numerical experiments of the proposed distributional off-policy evaluation method in several challenging tasks. We first describe some practical implementation of our approach in deep RL experiments. Then we present some numerical

studies in Atari game (Mnih et al., 2013) with image observations as states and a scalar reward. Finally, we use several challenging maze tasks to show the promising performance of our method in learning the distribution of a multivariate discounted cumulative reward. For the ease of presentation, we call the distribution of Y given any state-action pair under the target policy as value distribution and denote Wasserstein-GAN by WGAN.

5.1 Practical Implementation

WGAN Penalty Note that in Problem (14), we need to enforce the Lipschitz constraint for the discriminator h. Following the idea of WGAN with gradient penalty (Gulrajani et al., 2017*b*) that essentially used the exact penalty method (e.g., Chapter 9 of (Cui and Pang, 2021)), we construct a penalty function on the gradient norm of the discriminator h as

$$L_{\text{gp}}(h) = \lambda \frac{1}{NT} \sum_{i=1}^{N} \sum_{i=0}^{T-1} (\|\partial h\|_2 - 1)^2, \tag{21}$$

where λ is some hyper-parameter. We follow the principle of WGAN-GP and the penalty loss in Eq. (21) is optimized by the stochastic gradient descent with the objective function in Problem (14).

Overall Architecture The overall architecture of our algorithm for deep RL tasks is given in Fig. 1. We use the distributional Q-network as the generator, which contains a convolution block and two fully connected layers. The convolution block is the same as the DQN-network, and each fully connected layer contains 512 hidden units. The noise is sampled from uniform$(0, 1)$ and has the same dimension as the state features. The noise is integrated with the state features in fully connected layers. The generator finally outputs $|\mathcal{S}| \times |\mathcal{A}|$ units, where each prediction of $\widehat{\mathbb{G}}^{\pi}(\mathbf{Z}, s, a)$ is represented as a m-dimensional vector. Similar to the most value-based deep RL algorithms, we use a target generator to calculate the distributional Bellman target. The target generator has the same architecture

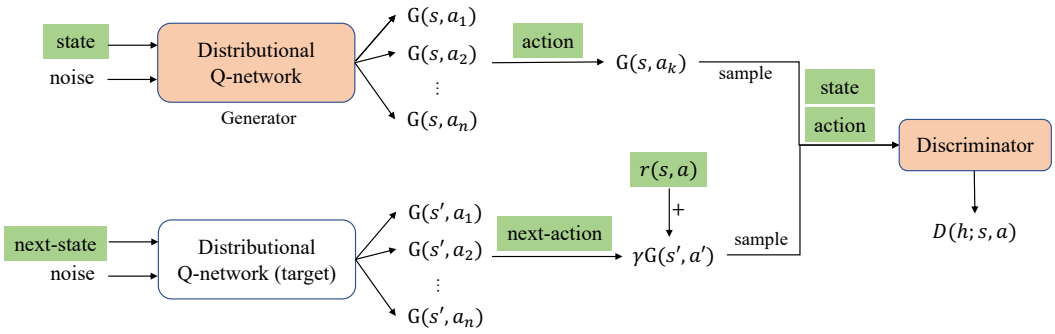


Figure 1: The overall architecture of our method. The orange parts indicate the generator and the discriminator that contain trainable parameters, and the blue elements indicate (s, a, r, s', a') sampled from an offline dataset or a replay buffer.

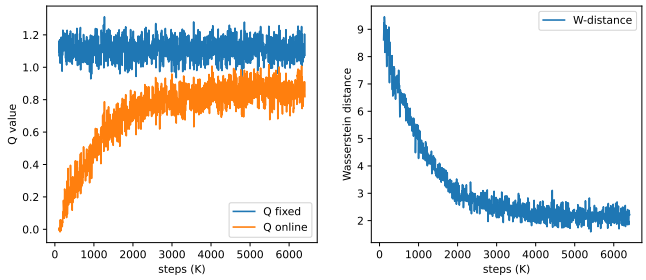


Figure 2: Policy evaluation in Atari task *Pong*. (*left*) The Q-value of the evaluated policy (Q-fixed) and the trainable distributional value function. (*right*) The W-distance between samples coming from the Q-online and Q-fixed.

as the main generator and is periodically synchronized with the main generator.

The architecture of the discriminator is similar to the generator. The discriminator takes (s, a) pair as the condition for output. The output random sample Y is encoded by a fully connected layer and concatenated with state-action features before the final output layer. The output of the discriminator is a linear unit that represents $\partial h(\widetilde{Y}, s, a)$, where $\widetilde{Y}$ can be sampled from the current value distribution. The discriminator is trained more frequently than the generator in order to provide reasonable stochastic gradients to the generator. We give the detailed hyper-parameters for training in the appendix.

5.2 Atari Experiments

We conduct a policy evaluation in Atari game *Pong* that has a single-dimensional reward function. We use an expert policy trained by a IQN network (Dabney, Ostrovski, Silver and

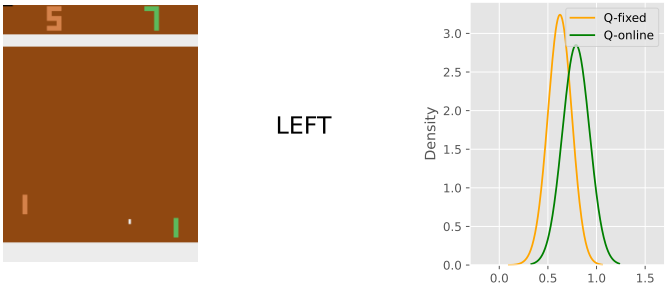
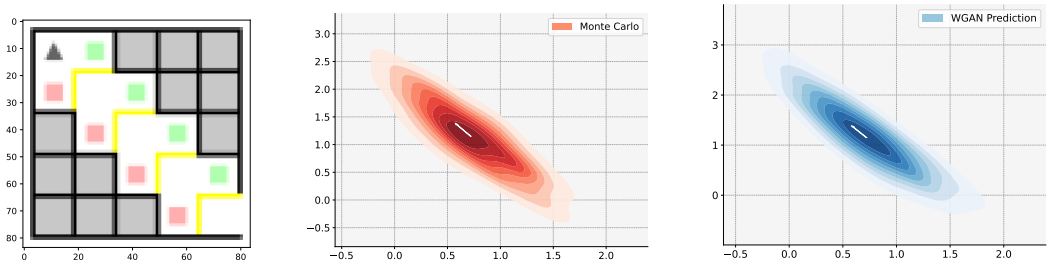


Figure 3: The state-action pair and the comparison of value distribution between Q-fixed and Q-online. The left and middle graphs of this figure are the corresponding state-action pair. The agent (i.e., green baffle) takes 'left' action to prevent the lose of the ball. The right one is the value distribution fitted by KDE.

Munos, 2018) for evaluation, which is denoted as 'Q-fixed'. 'Q-fixed' can be regarded as an oracle value distribution. We follow the expert policy mixed with an ϵ-greedy strategy to generate 10^5 examples as an offline dataset for policy evaluation.

We train the distributional Q-network for 7 million steps and record the average of value prediction for some randomly sampled batch (s, a) pairs (which is indeed Q-function/value). The result is given in Fig. 2. We denote the output by $\widehat{\mathbb{G}}(z, s, a)$ as "Q-online", where z is a random sample from uniform$(0, 1)$, and the oracle one as "Q-fixed". According to Fig. 2, the value of Q-function is small at the beginning since the Q-network is randomly initialized, but then it increases gradually to the oracle one. As we can see, with the update of Q-network, the value Q-function given by Q-online approaches Q-fixed asymptotically. We also observe that the supreme 1-Wasserstein distance between samples coming from the Q-online and Q-fixed decreases gradually. This is consistent with our theoretical results.

We also randomly select several sampled state-action pairs and report their corresponding value distribution predictions. For each state-action pair, we generate many random samples $z_i \sim$ uniform$(0, 1)$ as described in Algorithm 1 in the appendix and use the corresponding predictions $\widehat{\mathbb{G}}(z_i, s, a)$ to approximate the value distribution. Then we use kernel density estimation (KDE) to fit the predicted values. We present the value distribution of Q-fixed and Q-online in Fig. 3 (and the other two Fig. 1, 2 in the appendix). The value distribution estimation given by our algorithm is very close to the true one of the target policy.



(a) Maze 1 (b) Monte-Carlo estimation (c) WGAN prediction

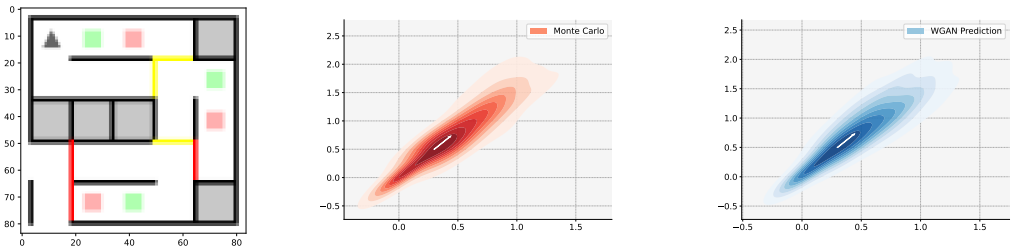
Figure 4: Maze 1 experiments. (a) The observation of Maze 1. (b) The Monte-Carlo estimation of value distributions. We draw 200 trajectories from Monte-Carlo evaluation and then estimate the distribution of their discounted cumulative rewards by KDE. (c) The WGAN-based prediction of the value distribution. The white arrows indicate the eigenvectors corresponding to the first principle component.

5.3 Maze Experiments with A Multivariate Reward

In this subsection, we conduct the task of policy evaluation in three maze environments. The observations of mazes are images that have the same dimensions as that of the Atari task used in the previous subsection, which has 84×84 image observations as states. In addition, the maze tasks considered here have a multivariate reward that is given as a vector at each time step. We impose either strongly negative or positive correlations among different coordinates of the reward function. Simply extending IQN to the multi-variate reward settings does not capture such relationships since it is only able to learn the marginal distribution of each dimension of the reward function independently. In contrast, our method will learn the correlation between reward dimensions since we jointly estimate the value distribution.

In all mazes, the agent is initially located in the triangle position. Different colors of squares in mazes indicate different sources of rewards (i.e., different coordinates of the reward function). We perform policy evaluation on a uniformly random policy. The Monte-Carlo method is conducted to compute the ground truth. We give the details of our simulation settings and evaluation results for each maze in the following.

Maze 1 shown in Fig. 4 has two sources of rewards (i.e., green and red). The yellow



(a) Maze 1 (b) Monte-Carlo estimation (c) WGAN prediction

Figure 5: Maze 2 experiments. (a) The observation of Maze 2. The red wall cannot be passed. (b) and (c) represent the value distributions of Monte-Carlo estimation and WGAN prediction. We sample 200 points from each distribution and fit the density through KDE. Our policy evaluation learns the correlation between the two dimensions in the reward function.

wall can only be passed in one way. If the agent obtains a particular reward (green or red) from one side, it will pass the wall with a high probability and cannot get the reward of another color on the other side. Thus, these two rewards in this maze are exclusive. As shown in Fig. 4, our method successfully learns the negative correlation between two sources of rewards (i.e., two-dimensional cumulative discounted rewards) and outputs similar predictions as those given by Monte-Carlo estimation.

Maze 2 also has two sources of rewards (i.e., green and red). Since these two reward sources are located close to each other, they are positively correlated. As shown in Fig. 5, our method again is able to learn the positive correlation between these two values (i.e., Y_1 and Y_2).

Learning value distribution related to Maze 3 is even more challenging. In this maze, we have four sources of rewards (i.e., orange, blue, green, red) with more complex correlations. In addition, when generating each source of rewards, we add one noisy variable, which follows uniform$(0, 0.2)$. The description and the corresponding results can be summarized as follows.

- The orange and blue rewards are positively correlated since they are located closely. The policy evaluation result given by our method successfully captures this relationship. See Fig. 6(b).

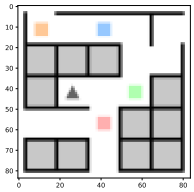
- The orange and green rewards have complex correlations. The agent starting from the triangle point has three possible paths. (i) If the agent goes down, it cannot obtain both rewards. (ii) If the agent goes right, it will obtain the green reward. Then the agent can go up and try to obtain the orange reward. If succeeded, the agent will obtain both rewards. (iii) After getting the green reward, the agent can go right and obtain no reward. The complex relationship is also discovered by our algorithm, as shown in Fig. 6(c).

- The orange and red rewards are negatively correlated. If the agent chooses to go down to obtain the red reward, it is hard to turn back to obtain the orange reward. The result is given in Fig. 6(d).

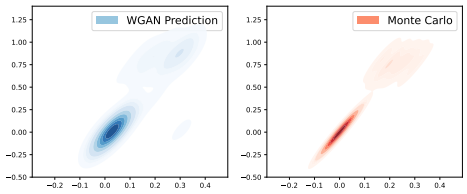
- The correlations of other kinds of rewards are similar to the examples described above.

According to our experimental results, overall our distributional policy evaluation method is capable of characterizing the joint distributional behavior of the multi-variate discounted cumulative reward in several challenging maze tasks. The value predictions align well with the Monte-Carlo estimation.

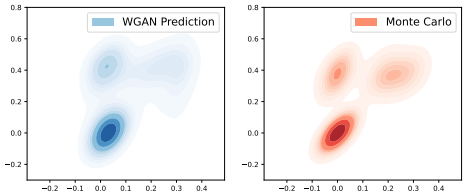
Lastly, for Maze 3, we add a different noisy variable (i.e., uniform(0, 0.1)) on each source of rewards and decrease the mean of the reward function by 0.1. Then we apply our method to learn the value distribution, which is given in Figure 7. The goal here is to demonstrate the necessity of learning a joint value distribution when comparing with other policies in some high-stake scenarios. For example, Figure 6 (c) and Figure 7 (a) can be understood as the distributional performance of two different policies. If one prefers a policy that maximizes the expected discounted cumulative reward, then obtaining orange and green rewards is better than getting orange and blue ones. Otherwise, if one is risk averse, obtaining orange and blue rewards may be more desirable. This example also demonstrates the potential of our method in the domain of the risk sensitive RL.



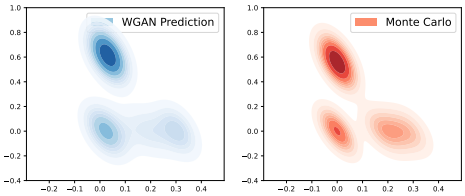
(a) Maze 3



(b) orange-blue

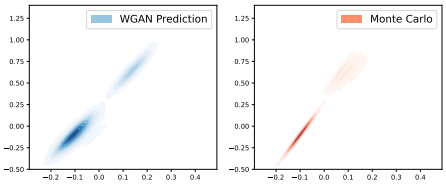


(c) orange-green

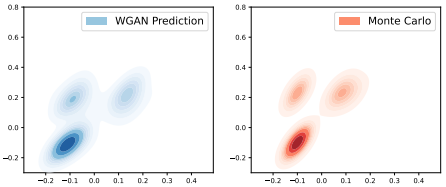


(d) orange-red

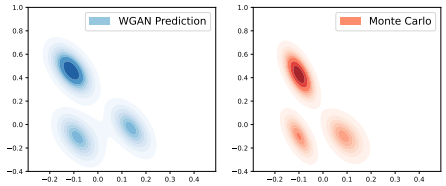
Figure 6: Maze 3 with an independent noisy variable (uniform$(0, 0.2)$) added on each reward function. (a) The observation of Maze 3. (b-d) The visualization of joint distributions between different pairs of dimensions. The value distribution of our algorithm aligned well with Monte-Carlo evaluation.



(a) orange-blue



(b) orange-green



(c) orange-red

Figure 7: Maze 3 with an independent noisy variable (uniform$(0, 0.1)$) added on each reward function. Meanwhile, we decrease all the rewards in previous experiment by 0.1. (a) The observation of Maze 3. (b-d) The visualization of joint distributions between different pairs of dimensions. We find (1) the value distribution has small shift ($\approx$0.1) in the space, and (2) has smaller contours and more concentrated distribution.

6 Conclusion

In this paper, we study the distributional off-policy evaluation problem in the batch reinforcement learning. We focus on estimating the joint distribution of a multivariate discounted cumulative reward by leveraging the batch data generated by some behavior policy. Based on the distributional Bellman equation, we propose an offline Wasserstein GAN-based approach to estimate the distribution given any target policy. Statistical theory such as finite sample error bound of our estimator to the truth in terms of the supreme 1-Wasserstein distance is established. Our theoretical results are appealing as the convergence rate is in terms of both N and T and does not require that data come from the stationary distribution. One possible extension of our work is to study the policy optimization for general objectives such as those three motivating examples discussed in Section 2.2.

References

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C. and Bellemare, M. (2021), 'Deep reinforcement learning at the edge of the statistical precipice', *Advances in Neural Information Processing Systems* **34**.

Ai, C. and Chen, X. (2003), 'Efficient estimation of models with conditional moment restrictions containing unknown functions', *Econometrica* **71**(6), 1795–1843.

Antos, A., Szepesvári, C. and Munos, R. (2008), Fitted q-iteration in continuous action-space mdps, *in* 'Advances in neural information processing systems', pp. 9–16.

Arjovsky, M., Chintala, S. and Bottou, L. (2017), Wasserstein generative adversarial networks, *in* 'International conference on machine learning', PMLR, pp. 214–223.

Bellemare, M. G., Dabney, W. and Munos, R. (2017), A distributional perspective on reinforcement learning, *in* 'International Conference on Machine Learning', PMLR, pp. 449–458.

Bellemare, M. G., Le Roux, N., Castro, P. S. and Moitra, S. (2019), Distributional reinforcement learning with linear function approximation, *in* 'The 22nd International Conference on Artificial Intelligence and Statistics', PMLR, pp. 2203–2211.

Ben-Tal, A. and Teboulle, M. (1986), 'Expected utility, penalty functions, and duality in stochastic nonlinear programming', *Management Science* **32**(11), 1445–1466.

Bertsekas, D. P. (1995), *Dynamic programming and optimal control*, Vol. 1, Athena scientific Belmont, MA.

Biau, G., Sangnier, M. and Tanielian, U. (2021), 'Some theoretical insights into wasserstein gans', *Journal of Machine Learning Research* .

Blundell, R., Chen, X. and Kristensen, D. (2007), 'Semi-nonparametric iv estimation of shape-invariant engel curves', *Econometrica* **75**(6), 1613–1669.

Chen, M., Liao, W., Zha, H. and Zhao, T. (2020), 'Statistical guarantees of generative adversarial networks for distribution estimation', *arXiv preprint arXiv:2002.03938* .

Chen, X. and Christensen, T. M. (2018), 'Optimal sup-norm rates and uniform inference on nonlinear functionals of nonparametric iv regression', *Quantitative Economics* **9**(1), 39–84.

Chen, Y., Gao, Q. and Wang, X. (2021), 'Inferential wasserstein generative adversarial networks', *arXiv preprint arXiv:2109.05652* .

Chow, Y., Ghavamzadeh, M., Janson, L. and Pavone, M. (2017), 'Risk-constrained reinforcement learning with percentile risk criteria', *The Journal of Machine Learning Research* **18**(1), 6070–6120.

Chung, K.-J. and Sobel, M. J. (1987), 'Discounted mdp?s: Distribution functions and exponential utility maximization', *SIAM journal on control and optimization* **25**(1), 49–62.

Cui, Y. and Pang, J.-S. (2021), 'Modern nonconvex nondifferentiable optimization'.

Dabney, W., Ostrovski, G., Silver, D. and Munos, R. (2018), Implicit quantile networks for distributional reinforcement learning, *in* 'International conference on machine learning', PMLR, pp. 1096–1105.

Dabney, W., Rowland, M., Bellemare, M. G. and Munos, R. (2018), Distributional reinforcement learning with quantile regression, *in* 'Thirty-Second AAAI Conference on Artificial Intelligence'.

Du, S., Krishnamurthy, A., Jiang, N., Agarwal, A., Dudik, M. and Langford, J. (2019), Provably efficient rl with rich observations via latent state decoding, *in* 'International Conference on Machine Learning', PMLR, pp. 1665–1674.

Ernst, D., Geurts, P. and Wehenkel, L. (2005), 'Tree-based batch mode reinforcement learning', *Journal of Machine Learning Research* **6**, 503–556.

Farahmand, A.-m., Ghavamzadeh, M., Szepesvári, C. and Mannor, S. (2016), 'Regularized policy iteration with nonparametric function spaces', *The Journal of Machine Learning Research* **17**(1), 4809–4874.

Frankle, J. and Carbin, M. (2018), 'The lottery ticket hypothesis: Finding sparse, trainable neural networks', *arXiv preprint arXiv:1803.03635* .

Fu, J., Norouzi, M., Nachum, O., Tucker, G., Wang, Z., Novikov, A., Yang, M., Zhang, M. R., Chen, Y., Kumar, A. et al. (2021), 'Benchmarks for deep off-policy evaluation', *arXiv preprint arXiv:2103.16596* .

Garcıa, J. and Fernández, F. (2015), 'A comprehensive survey on safe reinforcement learning', *Journal of Machine Learning Research* **16**(1), 1437–1480.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014), 'Generative adversarial nets', *Advances in neural information processing systems* **27**.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A. (2017*a*), 'Improved training of wasserstein gans', *arXiv preprint arXiv:1704.00028* .

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A. C. (2017*b*), Improved training of wasserstein gans, *in* 'Neural Information Processing Systems'.

Haas, M. and Richter, S. (2020), 'Statistical analysis of wasserstein gans with applications to time series forecasting', *arXiv preprint arXiv:2011.03074* .

Hansen, L. P. (1982), 'Large sample properties of generalized method of moments estimators', *Econometrica: Journal of the Econometric Society* pp. 1029–1054.

Hazan, E., Kakade, S., Singh, K. and Van Soest, A. (2019), Provably efficient maximum entropy exploration, *in* 'International Conference on Machine Learning', PMLR, pp. 2681–2691.

Hubbs, C. D., Perez, H. D., Sarwar, O., Sahinidis, N. V., Grossmann, I. E. and Wassick, J. M. (2020), 'Or-gym: A reinforcement learning library for operations research problems', *arXiv preprint arXiv:2008.06319* .

Jaquette, S. C. (1973), 'Markov decision processes with a new optimality criterion: Discrete time', *The Annals of Statistics* **1**(3), 496–505.

Jiang, N. and Li, L. (2016), Doubly robust off-policy value evaluation for reinforcement learning, *in* 'International Conference on Machine Learning', pp. 652–661.

Jin, C., Krishnamurthy, A., Simchowitz, M. and Yu, T. (2020), Reward-free exploration for reinforcement learning, *in* 'International Conference on Machine Learning', PMLR, pp. 4870–4879.

Kallus, N. and Uehara, M. (2019), 'Efficiently breaking the curse of horizon in off-policy evaluation with double reinforcement learning', *arXiv preprint arXiv:1909.05850* .

Kosorok, M. R. and Laber, E. B. (2019), 'Precision medicine', *Annual review of statistics and its application* **6**, 263–286.

Kress, R., Maz'ya, V. and Kozlov, V. (1989), *Linear integral equations*, Vol. 82, Springer.

Le, H., Voloshin, C. and Yue, Y. (2019), Batch policy learning under constraints, *in* 'International Conference on Machine Learning', pp. 3703–3712.

Levine, S., Kumar, A., Tucker, G. and Fu, J. (2020), 'Offline reinforcement learning: Tutorial, review, and perspectives on open problems', *arXiv preprint arXiv:2005.01643* .

Liao, P., Qi, Z. and Murphy, S. (2020), 'Batch policy learning in average reward markov decision processes', *arXiv preprint arXiv:2007.11771* .

Liu, J., Zhang, Y., Wang, X., Deng, Y. and Wu, X. (2019), 'Dynamic pricing on e-commerce platform with deep reinforcement learning', *arXiv preprint arXiv:1912.02572* .

Liu, Q., Li, L., Tang, Z. and Zhou, D. (2018), Breaking the curse of horizon: Infinite-horizon off-policy estimation, *in* 'Advances in Neural Information Processing Systems', pp. 5356–5366.

Lizotte, D. J., Bowling, M. H. and Murphy, S. A. (2010), Efficient reinforcement learning with multiple reward functions for randomized controlled trial analysis, *in* 'ICML'.

Lyle, C., Bellemare, M. G. and Castro, P. S. (2019), A comparative analysis of expected and distributional reinforcement learning, *in* 'Proceedings of the AAAI Conference on Artificial Intelligence', Vol. 33, pp. 4504–4511.

Mirza, M. and Osindero, S. (2014), 'Conditional generative adversarial nets', *arXiv preprint arXiv:1411.1784* .

Misra, D., Henaff, M., Krishnamurthy, A. and Langford, J. (2020), Kinematic state abstraction and provably efficient rich-observation reinforcement learning, *in* 'International conference on machine learning', PMLR, pp. 6961–6971.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. and Riedmiller, M. (2013), 'Playing atari with deep reinforcement learning', *arXiv preprint arXiv:1312.5602* .

Nachum, O., Chow, Y., Dai, B. and Li, L. (2019), Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections, *in* 'Advances in Neural Information Processing Systems', pp. 2315–2325.

Newey, W. K. and Powell, J. L. (2003), 'Instrumental variable estimation of nonparametric models', *Econometrica* **71**(5), 1565–1578.

Prashanth, L. and Ghavamzadeh, M. (2013), 'Actor-critic algorithms for risk-sensitive mdps'.

Precup, D. (2000), 'Eligibility traces for off-policy policy evaluation', *Computer Science Department Faculty Publication Series* p. 80.

Rockafellar, R. T., Uryasev, S. et al. (2000), 'Optimization of conditional value-at-risk', *Journal of risk* **2**, 21–42.

Roijers, D. M., Vamplew, P., Whiteson, S. and Dazeley, R. (2013), 'A survey of multi-objective sequential decision-making', *Journal of Artificial Intelligence Research* **48**, 67–113.

Rowland, M., Bellemare, M., Dabney, W., Munos, R. and Teh, Y. W. (2018), An analysis of categorical distributional reinforcement learning, *in* 'International Conference on Artificial Intelligence and Statistics', PMLR, pp. 29–37.

Rust, J. (1987), 'Optimal replacement of gmc bus engines: An empirical model of harold zurcher', *Econometrica: Journal of the Econometric Society* pp. 999–1033.

Schmidt-Hieber, J. (2020), 'Nonparametric regression using deep neural networks with relu activation function', *The Annals of Statistics* **48**(4), 1875–1897.

Shen, Y., Tobia, M. J., Sommer, T. and Obermayer, K. (2014), 'Risk-sensitive reinforcement learning', *Neural computation* **26**(7), 1298–1328.

Shi, C., Wan, R., Chernozhukov, V. and Song, R. (2021), 'Deeply-debiased off-policy interval estimation', *arXiv preprint arXiv:2105.04646* .

Shi, C., Zhang, S., Lu, W. and Song, R. (2020), 'Statistical inference of the value function for reinforcement learning in infinite horizon settings', *arXiv preprint arXiv:2001.04515* .

Sutton, R. S. and Barto, A. G. (2018), *Reinforcement learning: An introduction*, MIT press.

Tang, Z., Feng, Y., Li, L., Zhou, D. and Liu, Q. (2019), 'Doubly robust bias reduction in infinite horizon off-policy estimation', *arXiv preprint arXiv:1910.07186* .

Thomas, P. and Brunskill, E. (2016), Data-efficient off-policy policy evaluation for reinforcement learning, *in* 'International Conference on Machine Learning', pp. 2139–2148.

Thomas, P. S., Theocharous, G., Ghavamzadeh, M., Durugkar, I. and Brunskill, E. (2017), Predictive off-policy policy evaluation for nonstationary decision problems, with applications to digital marketing, *in* 'Twenty-Ninth IAAI Conference'.

Uehara, M., Huang, J. and Jiang, N. (2020), Minimax weight and q-function learning for off-policy evaluation, *in* 'International Conference on Machine Learning', PMLR, pp. 9659–9668.

Wang, H. and Zhou, X. Y. (2020), 'Continuous-time mean–variance portfolio selection: A reinforcement learning framework', *Mathematical Finance* **30**(4), 1273–1308.

Wang, L., Zhou, Y., Song, R. and Sherwood, B. (2018), 'Quantile-optimal treatment regimes', *Journal of the American Statistical Association* **113**(523), 1243–1254.

Xie, T., Ma, Y. and Wang, Y.-X. (2019), 'Towards optimal off-policy evaluation for reinforcement learning with marginalized importance sampling', *arXiv preprint arXiv:1906.03393* .

Xie, Y., Liu, B., Liu, Q., Wang, Z., Zhou, Y. and Peng, J. (2018), 'Off-policy evaluation and learning from logged bandit feedback: Error reduction via surrogate policy', *arXiv preprint arXiv:1808.00232* .

Yang, D., Zhao, L., Lin, Z., Qin, T., Bian, J. and Liu, T.-Y. (2019), 'Fully parameterized quantile function for distributional reinforcement learning', *Advances in neural information processing systems* **32**, 6193–6202.

Zhang, J., Koppel, A., Bedi, A. S., Szepesvari, C. and Wang, M. (2020), 'Variational policy gradient method for reinforcement learning with general utilities', *arXiv preprint arXiv:2007.02151* .

Zhang, R., Dai, B., Li, L. and Schuurmans, D. (2020), Gen{dice}: Generalized offline estimation of stationary values, *in* 'International Conference on Learning Representations'. **URL:** *https://openreview.net/forum?id=HkxlcnVFwB*

Zhang, S., Liu, B. and Whiteson, S. (2020), Gradientdice: Rethinking generalized offline estimation of stationary values, *in* 'International Conference on Machine Learning', PMLR, pp. 11194–11203.

Zhong, H., Fang, E. X., Yang, Z. and Wang, Z. (2020), 'Risk-sensitive deep rl: Variance-constrained actor-critic provably finds globally optimal policy', *arXiv preprint arXiv:2012.14098* .

Zolotarev, V. M. (1976), 'Metric distances in spaces of random variables and their distributions', *Mathematics of the USSR-Sbornik* **30**(3), 373.